

## Problem 4—Asteroids

Atari's "Asteroids" hit the video arcade market in 1979, quickly becoming a huge success. The game display was not based on pixels. Rather, the hardware supported native rendering of vector graphics directly onscreen using an electron beam. Both the asteroids and the player's ship would wraparound when they reached the edges of the screen. Unlike the old arcade version of asteroids, your version will be pixel based, or more accurately, ASCII-based.

### INPUT SPECIFICATION.

You will be given a list of asteroids to render, each specified by an x coordinate, y coordinate, and asteroid type, in that order. The x and y coordinates are integers in the range [0, 19], and the asteroid type is an integer in the range [1, 3]. The integers are delimited by a single space. Coordinate (0,0) is the upper left hand corner. Each asteroid will be followed by <EOLN>. You may assume that the asteroids are given in such a way that they do not overlap graphically, but they may come very close.

### OUTPUT SPECIFICATION.

Render the given asteroids on a 20 by 20 display with horizontal and vertical wraparound. The x and y coordinates of an asteroid correspond with its upper-left corner. Remember, format counts, your output should exactly match the correct output. Notice that the horizontal parts are hyphens, not underscores!

Asteroid type 1

```
 /-\
 | |
 | | \---\
 | |   |
 \-----/
```

Asteroid type 2

```
 \---\
  \  \
   \  \
    \---\
```

Asteroid type 3

```
 /--\
 | /
 | \
 \--/
```

### SAMPLE INPUT

```
0 0 1<EOLN>
8 18 3<EOLN>
3 10 2<EOLN>
18 10 2<EOLN>
<EOF>
```

### SAMPLE OUTPUT

```
 /-\.....|. \.....<EOLN>
|. |.....\--/.....<EOLN>
|. \---\.....<EOLN>
|.....|.....<EOLN>
\-----/.....<EOLN>
.....<EOLN>
.....<EOLN>
.....<EOLN>
.....<EOLN>
.....<EOLN>
--\ \---\.....\<EOLN>
.. \ \.....\<EOLN>
\.....\ \.....\<EOLN>
. \---\ \---\.....<EOLN>
.....<EOLN>
.....<EOLN>
.....<EOLN>
.....<EOLN>
...../--\.....<EOLN>
.....|. /.....<EOLN>
<EOF>
```